Refine Search

Search Results -

Terms	Documents
L16 and L6	1

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L17

Refine Search

Recall Text | Clear | Interrupt

Search History

DATE: Monday, December 06, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L17	L16 and l6	1	L17
L16	L1 same repair	348	L16
L15	L14 same repair	1	L15
L14	second near3 signature	2465	L14
L13	L6 same repair	1	L13
L12	L11 and signature	4	L12
L11	6085334[uref]	27	L11
L10	6085334[pn]	2	L10
L9	L8 and l6	2	L9
L8	bisr or bist	3186	L8
L7	L6 same test$	10	L7
L6	l1 same signature	170	L6
L5	L1 and l2	1	L5
L4	L1 and bist	62	L4

L3	L2 same l1	0	L3
L2	repair same bist	141	L2
L1	augmented	47110	L1

END OF SEARCH HISTORY

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#
End of Result Set

Generate Collection Print

L5: Entry 1 of 1 File: USPT Jul 3, 2001

DOCUMENT-IDENTIFIER: US 6255836 B1
TITLE: Built-in self-test unit having a reconfigurable data retention test

Drawing Description Text (3):
FIG. 1 is a functional block diagram of a memory equipped with an augmented BIST unit according to one embodiment; and

Drawing Description Text (4):
FIG. 2 is a functional block diagram of a second embodiment of a memory with an augmented BIST unit.

Detailed Description Text (2):
In the following description, the terms "assert" and "de-assert" are used when discussing logic signals. When a logic signal is said to be asserted, this indicates that an active-high signal is driven high, whereas an active-low signal is driven low. Conversely, de-assertion indicates that an active-high signal is driven low, and that an active-low signal is driven high. As used herein, the term "BIST" refers to the actual test, while "BIST unit" and "BIST circuitry" refer to the circuitry that performs BIST. Similarly, "BISR" refers to the process of built-in self repair, while "BISR unit" and "BISR circuitry" refer to the circuitry that performs BISR.

Detailed Description Text (6):
When the TEST line is asserted, the BIST unit 110 takes control of the ADDR, R/W, and DATA lines, and conducts a pattern of read and write operations designed to detect faults in the memory array 112. During write operations, the BIST unit 110 supplies test data to the memory array on the DATA lines. During read operations, the BIST unit 110 compares values on the DATA lines to the expected output. When a mismatch is detected, BIST unit 110 asserts the FAIL signal line. In alternative embodiments, BIST unit 110 may instead notify a built-in self-repair (BISR) unit of the error, and the BISR unit may replace the faulty memory location with a redundant memory location.

Previous Doc Next Doc Go to Doc#

L9: Entry 1 of 2 File: USPT Dec 30, 2003

DOCUMENT-IDENTIFIER: US 6671839 B1
TITLE: Scan test method for providing real time identification of failing test patterns and test bist controller for use therewith

Detailed Description Text (3):
While the present invention is described herein with reference to a built-in self-test (BIST) controller, it is to be understood that the method of the present invention applies to any scan testing method which uses a signature register and not only to BIST methods. Deterministic scan vectors (as opposed to pseudo-random patterns generated by a PRPG) can be scanned into memory elements and use a signature register when scanning out.

Detailed Description Text (8):
With reference to FIG. 3, a test controller according to the present invention is augmented by the provision of an expected signature register 50, also referred to herein as a shadow register, having a serial input 52, a serial output 54 and a plurality of memory elements which define a bit length which is at least equal to the bit length of the signature register. The memory elements of the shadow register are configurable in Hold mode for holding their contents constant and in Shift mode for shifting data through the register. The role of the shadow register is to receive and hold the expected signature of test patterns which are being executed, and, when the test patterns have completed executing, to receive the actual signature from the signature register during a swapping operation in which the expected signature is moved into the signature register and the actual signature is moved into the shadow register. The expected signature is loaded into the shadow register while the corresponding test patterns are being executed. The Hold facility is also required when the test patterns are executed at one clock rate and data is shifted through the shadow register at a different, usually slower, clock rate and for holding an expected signature until required. Optionally, an additional memory element 56 can be added to the shadow register to store a "start bit" to improve the diagnosability of the circuit.

Detailed Description Text (23):
Typically, the test blocks are performed under control of a first clock, usually operating at the application or design speed (also called system speed) of the core block, while the loading and unloading of the shadow register contents is performed under control of a second clock, usually a slower test clock. The goal of the method is to maximize the number of times the signature register can be updated without interrupting the application of test patterns to the circuit. Ideally, the signature can be updated after each trial. A trial refers to the loading of a test pattern, capturing the circuit response and unloading the responses into the MISR. In order to achieve this goal, there must be enough clock cycles of the second clock to scan in all the bits of an expected signature. Because a start bit is needed for synchronization (i.e. to start counter 64 and other functions in FSM 58), the MISR width plus one is the number of second clock cycles that must fit within a single trial. If a clock ratio (i.e., the clock rate of the first clock divided by the second clock) of four is assumed, and a maximum scan chain length of 512 is tested by the logic BIST controller, the trial duration is 128 (512/4) clock cycles when expressed in terms of the second clock. This number of clock cycles is sufficient to accommodate an average sized MISR of 24 or 32 bits. For higher system

clock frequencies, the clock ratio might need to be increased to 8 or 16. For a clock ratio of 16, there are enough second clock cycles for a 24-bit MISR, but not for a 32 bit MISR.

Previous Doc Next Doc Go to Doc#

L12: Entry 1 of 4 File: USPT Jul 20, 2004

DOCUMENT-IDENTIFIER: US 6766468 B2
TITLE: Memory BIST and repair

Brief Summary Text (11):
In accordance with the teachings herein, the present invention provides a method of storing memory redundancy allocation signatures for memory elements, such as memory blocks or banks, on-chip. Pursuant to the method, each memory element capable of redundancy on the chip is interrogated, one at a time using a built-in self test program. A redundancy allocation signature is then stored on the chip for each memory element from said built-in self test program, and redundancy is enabled for each memory element by interpreting the stored redundancy allocation signature for that memory element.

Detailed Description Text (3):
Each memory block is testable via a BIST engine and contains redundancy block(s). The BIST engine interrogates each 1 Meg memory block, one at a time, and produces a redundancy allocation signature based upon the interrogation which describes an optimized memory repair for that memory block designating replacement redundant memory cells, columns, rows, etc. to be substituted for faulty memory cells, columns, rows, etc. in the memory block. The redundancy allocation signature is then shifted into the FAR (Fuse Address Register). The redundancy allocation signature provides redundancy allocation information for the failing portions of each 1 Meg memory block.

Detailed Description Text (5):
Memory block3 is tested first by the BIST engine. The redundancy allocation signature for memory block3 is then stored in the FAR and then shifted into the FL0 (Fuse Latch0).

Detailed Description Text (6):
Memory block2 is then interrogated using the same process. The redundancy allocation signature for memory block2 is stored in the FAR and is then shifted to the FL0, and the contents of the FL0 (memory block3 redundancy allocation signature) is shifted into FL1.

Detailed Description Text (7):
Memory block1 is interrogated next. The redundancy allocation signature for memory block1 is stored in the FAR and is then shifted into the FL0, the contents of the FL0 (memory block2 redundancy allocation signature) is shifted into the FL1, and the contents of the FL1 (memory block3 redundancy allocation signature) is shifted into the FL2.

Detailed Description Text (8):
Memory block0 is interrogated last. The redundancy allocation signature for memory block0 is stored in the FAR and then shifted into the FL0. The contents of the FL0 (memory block1 redundancy allocation signature) is shifted into the FL1, the contents of the FL1 (memory block2 redundancy allocation signature) is shifted into the FL2, and the contents of the FL2 (memory block3 redundancy allocation signature) is shifted into the FL3.

Detailed Description Text (10):
At this point, each memory block has been interrogated by the BIST engine, and each block's redundancy allocation signature is stored in its respective fuse latch FL. Each fuse latch FL can then be accessed and used to enable redundancy allocation or to steer fuse-blow current/voltage for fuse activation (i.e. electronic fuse blow) on each Memory block.

Detailed Description Text (12):
The present invention provides the ability to store memory redundancy allocation signature on-chip in existing storage elements.

US Reference Patent Number (12):
6085334

CLAIMS:

1. A method of storing memory redundancy allocation signatures for separate memory blocks of a memory, on-chip, comprising: interrogating each separate memory block, wherein each separate memory block comprises a plurality of memory cells arranged in a plurality of rows and columns, capable of redundancy on a chip, one memory block at a time, using a built-in self test (BIST) engine; providing a separate fuse storage for each separate memory block and a single fuse address register (FAR) which is shared between the separate memory blocks during testing by the BIST engine, wherein each memory block is tested by the BIST engine, and a redundancy allocation signature for that memory block is stored in the single fuse address register (FAR) and is then shifted into the separate fuse storage for that memory block, to store on the chip a redundancy allocation signature for each memory block from said built-in self test program; and enabling redundancy for each memory block by interpreting the stored redundancy allocation signature for that memory block.

3. The method of claim 1, including storing the redundancy allocation signature for each memory block in a separate fuse storage comprising a separate fuse latch for that memory block.

4. The method of claim 3, wherein: memory blockn is tested first by the BIST engine, and the redundancy allocation signature for memory blockn is stored in the single fuse address register and is then shifted into a fuse latch0; one or more intermediate memory blocks are tested next by the BIST engine, and the redundancy allocation signature for each intermediate memory block is stored in the single fuse address register, and is then shifted to the fuse latch0, and the contents of the fuse latch0 is shifted into an intermediate latch; memory block0 is tested last by the BIST engine, and the redundancy allocation signature for memory block0 is stored in the single fuse address register and is then shifted into the fuse latch0, the contents of the fuse latch0, which is an intermediate memory block redundancy allocation signature, is shifted into an intermediate fuse latch, and the contents of an intermediate fuse latch, which is the memory blockn redundancy allocation signature, is shifted into the fuse latchn.

Previous Doc Next Doc Go to Doc#

L12: Entry 2 of 4 File: USPT Nov 18, 2003

DOCUMENT-IDENTIFIER: US 6651202 B1
TITLE: Built-in self repair circuitry utilizing permanent record of defects

Brief Summary Text (16):
In order to enhance the repair process, on-chip built-in self repair (BISR) circuitry for repairing faulty memory cells has evolved. BISR circuitry functions internal to the integrated circuit without detailed interaction with external test or repair equipment. In the typical BISR approach, suitable test algorithms developed and implemented in BIST or BIST-like circuitry. These test patterns may be capable of detecting stuck-at, stuck-open, bridging faults and retention faults during memory tests. Following execution of the test patterns, the BISR circuitry analyzes the BIST "signature" (results) and, in the event of detected faults, automatically reconfigures the defective memory utilizing redundant memory elements to replace the defective ones. A memory incorporating BISR is therefore defect-tolerant. The assignee of the present invention, LSI Logic Corporation, has addressed different methods of repairing faulty memory locations utilizing BIST and BISR circuitry, as disclosed in U.S. Pat. No. 5,764,878, entitled "BUILT-IN SELF REPAIR SYSTEM FOR EMBEDDED MEMORIES", U.S. patent application No. 09/209,938, entitled "REDUNDANCY ANALYSIS FOR EMBEDDED MEMORIES WITH BUILT-IN SELF TEST AND BUILT-IN SELF REPAIR" filed Dec. 11, 1998, now U.S. Pat. No. 6,067262, and U.S. patent application No. 09/209,996, entitled "TESTING SCHEME FOR EMBEDDED MEMORIES USING BISR AND FUSE ID" filed Dec. 11, 1998, now U.S. Pat. No. 6,367,042, all of which are hereby incorporated by reference as if set forth in their entirety.

Brief Summary Text (22):
In the disclosed embodiment of the invention, an integrated circuit die of a semiconductor wafer is provided with BIST/BISR circuitry and an embedded memory or similar circuit. The integrated circuit also includes a fuse array or other none-volatile circuitry capable of storing address information for defective memory locations. During manufacture, the integrity of the embedded memory of each integrated circuit die is preferably tested under a variety of conditions (also referred to as stress factors) via the BIST/BISR circuitry. The results of these tests are stored and compiled in ATE. The results are also referred to as BIST signatures or memory repair solutions. If the repair solutions indicate that the embedded memory is repairable, the on-chip fuse array of the integrated circuit is programmed with information indicative of all of the detected defective memory locations. Programming of the fuse array may occur prior to or following singulation and packaging of the integrated circuit die.

Detailed Description Text (7):
On-chip BIST structures such as the BIST state machine/controller 102 typically communicate with external devices via an IEEE 1149.1 compliant interface. When implemented with this interface, IEEE 1149.1 test access port (TAP) and boundary scan structures (not shown) are also present on the integrated circuit IC. When implemented in such a fashion, a simple command sequence initiates BIST operation. After the BIST circuitry has completed its test patterns, the BIST state machine/controller 102 scans the results to off-chip test equipment via the test access port. In the disclosed embodiment of the invention, the BIST "signature" can also be retrieved via an output of the FLARESCAN register 114 as discussed more fully below.

Detailed Description Text (22):
Beginning with FIG. 3A, the exemplary fuse array 150 programming procedure is described in greater detail. Following commencement of the programming procedure (step 200) for a given integrated circuit IC, the integrated circuit IC is exposed to an initial set of one or more stress factors in step 202. As an example, the integrated circuit IC may be subjected to a minimum temperature as specified in the data sheet of the device. A predetermined amount of time is preferably allowed to elapse in order for the integrated circuit IC to adjust to the initial set of stress factors. Power is also applied to the integrated circuit IC at this stage. Next, an initial BIST/BISR routine is executed at step 204. In the disclosed embodiment of the invention, diagnosis of the integrated circuit IC by the BIST/BISR circuitry 122 is performed upon an initial power-up or when initiated by the external ATE 160. In conjunction with step 204, an initial set of faulty memory locations for the integrated circuit IC under test is developed in step 206. This initial set of faulty memory locations (e.g., the BIST signature), as well as any other desired information, is then retrieved and stored by the external ATE 160.

US Reference Patent Number (30):
6085334

Previous Doc Next Doc Go to Doc#

L12: Entry 3 of 4 File: USPT Jan 21, 2003

DOCUMENT-IDENTIFIER: US 6510398 B1
TITLE: Constrained signature-based test

Abstract Text (1):
A test system for structurally testing an integrated circuit device includes a pattern generator for generating successive random data patterns (scan chain). The test system further includes a constraint checker and corrector module, coupled to the pattern generator, to replace undesirable random data patterns (state elements joined together in the scan chain such that one state element is connected to a ground and the other state element is connected to a power supply) with desirable bit sequences to eliminate bus contention problems in the generated random data patterns. The test system further includes the integrated circuit device to be tested. The integrated circuit device receives the constrained random data patterns from the constraint checker and corrector module and outputs a test result. The test system further includes an X-masking module coupled to the integrated circuit device. The X-masking module receives the test result from the integrated circuit device, and it masks the test result by replacing unpredictable bit values (these are bit values generated due to not scanning some state elements in the scan chain) in the test result with predictable bit values. A signature analyzer coupled to the X-masking module receives the masked test result and compress the test result into a signature. Then a comparator coupled to the signature analyzer compares the signature with a predetermined test result to determine the functionality of the integrated circuit device.

Brief Summary Text (7):
BIST also uses signature analyzers to compress the test results into a single, smaller pattern or "signature" to reduce the amount of tester memory and circuitry required. The signature is then analyzed to determine whether the DUT is free of structural defects. For BIST, the random pattern generator and the signature analyzer circuitry are both built right into the DUT itself. This eliminates the need for using an external tester to test the DUT.

Detailed Description Text (3):
Embodiments of the present invention provide a method and apparatus for structurally testing DUTs to avoid the bus contention problems and the X-generation problems generally associated with BIST methodology of testing the integrated circuit devices. For one embodiment, a pattern generator, a constraint checker and corrector module, an X masking module, and a signature analyzer are incorporated into an external tester to circumvent the bus contention and the X-generation problems associated with the BIST methodology of testing DUTs. In another embodiment, the X masking module and the signature analyzer are incorporated into an integrated circuit die to circumvent the bus contention and the X-generation problems.

Detailed Description Text (9):
Action 170 in the method includes compressing the X-masked test result to output a single, smaller pattern or "signature" of the test result. In one embodiment, the signature can be one or more bit values. For example, the test result may include a number of 24-bit sequences of data, which could be compressed into a single, 12-24 bit signature. Generally, the bit values can be in the range of about 12 to 64.

Then the next step 175 in the process involves outputting the signature.

Detailed Description Text (10):
Action 180 in the method includes comparing the outputted signature with a predetermined test result. The predetermined test result is a calculated response. In one embodiment, the predetermined test result is one or more predetermined bit values. It can also be envisioned, that the action 180 be performed after the action 160. In this embodiment, the action 180 includes comparing the masked test result with the predetermined test result.

Detailed Description Text (16):
A signature analyzer 250 is coupled to the X-masking module 240 to receive the masked test result from the X-masking module 240 and compress the received masked test result (generally millions of bits of data) and output a signature of the test result. In one embodiment, the signature can be one or more bit values. For example, the test result may include a number of 24-bit sequences of data, which could be compressed into a single, 12-bit signature. Generally, the bit values can be in the range of about 12 to 64 bits. In one embodiment, the signature analyzer 250 is formed using a plurality of shift register latches. In another embodiment, the signature analyzer 250 is a programmable logic device incorporated into the external tester 280. The signature analyzer 250 and the X-masking module 240 can be incorporated as a single module into the external tester 280. By incorporating the X-masking module and the signature analyzer 250 into an external tester 280, the size of the die can become smaller, which can in turn reduce the design effort and hence reduce the cost of designing the integrated circuit device. It can also be envisioned that the external tester including the X-masking module 240 and the signature analyzer 250 can be incorporated into a personal computer. Also, it can be envisioned that the X-masking module and the signature analyzer can be incorporated into the die itself. This can significantly increase the size of the integrated circuit die.

Detailed Description Text (17):
A comparator 260 is coupled to the signature analyzer 250 to receive the signature from the signature analyzer 250. The comparator 260 compares the received signature with a predetermined test result and classifies the functionality of the DUT 230 based on an outcome of the comparison. The predetermined test result can be a calculated response. In one embodiment, the predetermined response can be one or more predetermined bit values. The system for structurally testing an integrated circuit device 200 further includes a memory 270 coupled to the constraint checker and corrector module 220 and to the signature analyzer 250 to store test algorithm instructions and the test results.

Detailed Description Text (18):
The above-described test method and apparatus provide, among other things, a test system that reduces the bus contention problems and the X-generation problems generally associated with the BIST methodology, and that yet retains advantages of the BIST methodology for structurally testing an integrated circuit device. In one embodiment, this is accomplished by incorporating a pattern generator, a constraint checker and corrector module, an X-masking module, and a signature analyzer into an external tester to eliminate bus contention problems and X-generation problems.

US Reference Patent Number (13):
6085334

CLAIMS:

5. The method of claim 4, which further comprises: masking the received test result by replacing unpredictable bit values in the test result with predictable bit values to control the outcome of the test result to a predictable test result; compressing the masked test result to a signature of the test result; and

outputting the signature.

6. The method of claim 5, which further comprises: comparing the signature with a predetermined test result.

9. A method of structurally testing an integrated circuit device comprising: generating random data patterns for testing the integrated circuit device; inspecting the generated random data patterns for undesirable random data patterns that can cause bus contention problems; separating the undesirable random data patterns found during the inspection of the generated random data patterns; replacing the separated undesirable random data patterns with desirable bit sequences to obtain constrained random data patterns; inputting the constrained random data patterns into the integrated circuit device for testing the integrated circuit device; testing the integrated circuit device using the inputted constrained random data patterns; receiving a test result; masking the received test result by replacing unpredictable bit values in the test result with predictable bit values to control the outcome of the test result to a predictable test result; compressing the masked test result to a signature of the test result; outputting the signature; comparing the signature with a predetermined test result; and determining whether the integrated circuit device is functioning properly based on an outcome of the comparison.

10. A method of structurally testing an integrated circuit device comprising: generating random data patterns for testing the integrated circuit device; constraining undesirable random data patterns in the generated random data patterns with desirable bit sequences; inputting the constrained random data patterns into the integrated circuit device for testing the integrated circuit device; testing the integrated circuit device using the inputted constrained random data patterns; receiving a test result; inspecting the test result for unpredictable bit values generated due to X-generation problems in the test result; separating the unpredictable bit values found during the inspection of the test result for the unpredictable bit values; replacing the separated unpredictable bit values in the test result with predictable bit values to obtain a masked test result; compressing the masked test result to a signature of the test result; outputting the signature; comparing the signature with a predetermined test result; and determining whether the integrated circuit device is functioning properly based on an outcome of the comparison.

13. A method of testing an integrated circuit device comprising: generating constrained random data patterns by constraining undesirable random data patterns in generated random data patterns with desirable bit sequences; inputting the constrained random data patterns into the integrated circuit device; testing the integrated circuit device using the constrained random data patterns; receiving a test result; inputting the received test result into a signature analyzer; compressing the test result by masking unpredictable bit values with predictable bit values in the test result and outputting a signature of the test result; comparing the signature to a predetermined test result; and determining whether the integrated circuit device is functioning properly based on an outcome of the comparison.

16. The method of claim 13, wherein the signature comprises one or more bit values.

19. A system for structurally testing an integrated circuit device comprising: an external pattern generator to generate successive random data patterns; an external constraint checker and corrector module, coupled to the pattern generator, to constrain nodes/elements in the random data pattern generated by the pattern generator; an integrated circuit device including circuitry, coupled to the constraint checker and corrector module, to receive the constrained random data pattern and to output a test result; an X-masking module, coupled to the integrated

circuit device, to mask the outputted test result from the integrated circuit device by replacing any unpredictable bit values with predictable bit values in the test result; and a signature analyzer, coupled to the X-masking module, to compress the masked test result into a signature.

20. The system of claim 19, which further comprises a comparator, coupled to the signature analyzer, to compare the signature to a predetermined test result and to classify the integrated circuit device based on an outcome of the comparison.

23. The system of claim 19, wherein the pattern generator, the constraint checker and corrector module, the X-masking module, and the signature analyzer comprise programmable logic devices.

27. The system of claim 19, wherein the signature analyzer and the X-masking module are formed from a plurality of shift register latches.

28. The system of claim 19, wherein the signature analyzer comprises a serial signature analyzer.

Previous Doc Next Doc Go to Doc#

L12: Entry 4 of 4 File: USPT Dec 17, 2002

DOCUMENT-IDENTIFIER: US 6496946 B2
TITLE: Electronic control apparatus with memory validation and method

Detailed Description Text (9):
FIG. 3 depicts a block diagram of memory validation logic 214 depicted in FIG. 2. Memory validation logic 214 includes a register array 300 of user programmable registers bi-directionally connected to the ADDRESS, DATA BUSES and to a validation logic 302. Validation logic 302 receives the control signal FREE CYCLE and generates the control signals LATCH DATA, FREEZE, SIGNATURE READY, and SIGNATURE FAILED. Validation logic 302 is more fully described below in connection with FIG. 5.

Detailed Description Text (10):
A signature generation unit 304 receives the data portion of ADDRESS, DATA BUSES, a control signal LATCH DATA, and the contents of a CONTROL/STATUS register in register array 300. The output of signature generation unit 304 is correlated to the data retrieved from the memory via the data portion of the ADDRESS, DATA BUSES. In the depicted embodiment, signature generation unit 304 is a linear feedback shift register (LFSR) instantiating a certain order polynomial function. One skilled in the art can select the particular polynomial order depending upon the circuit size budget and the required accuracy of the polynomial. In other embodiments, the signature generation unit could detect failures in non-volatile memory 212 using other algorithms. For instance, signature generation unit 304 could compare the parity of a datum (odd or even number of one's in the datum) with a stored parity bit. If the two parity results differed, then signature generation unit 304 would generate an error signal. This signal could be maintained as a "sticky bit." A sticky bit is a control signal that is the logical OR of a value of a function and the value of the function at a prior time. In this case, the function would be the pass/fail parity indication during the current cycle and the pass/fail parity indication of all previous memory accesses. The control bit will "stick" once a single parity error occurs. Other, more complicated techniques such as error correcting code algorithms could also be used.

Detailed Description Text (11):
A counter 306 also receives the control signal LATCH DATA and is bi-directionally coupled to validation logic 302. A first comparator 308 receives the address portion of ADDRESS, DATA BUSES and the contents of a STOP ADDRESS register in register array 300. An output of comparator 308 is input to validation logic 302. A second comparator 310 receives an output of the signature generation unit 304 and the contents of a CHECKSUM register in register array 300. An output of comparator 310 is also input to validation logic 302.

Detailed Description Text (12):
The specific operation of validation logic 302 and memory validation logic 214 is described below in connection with FIG. 5. That description is best understood in light of the following register and signal descriptions. Register array 300 includes six user programmable registers: START ADDRESS REGISTER contains the first memory address of a block of contiguous memory address to be tested. STOP ADDRESS

REGISTER contains the last memory address of a block of contiguous memory address to be tested. CONTROL/STATUS REGISTER contains various mode control bits and status flags. CONTROL/STATUS REGISTER is more fully described below in connection with FIG. 4. CHECKSUM REGISTER contains the expected signature for the array under test. CURRENT ADDRESS REGISTER contains the next memory address to be accessed in a validation operation. SIGNATURE REGISTER contains the result of the signature generation unit after a validation operation.

Detailed Description Text (13):
Validation logic 302 receives or generate the following five control signals: FREE CYCLE is generated by SIM 204. When asserted, FREE CYCLE indicates that CPU 1 is not using ADDRESS, DATA BUSES on the next access cycle. When de-asserted, CPU 1 will use the next bus cycle. LATCH DATA is generated by validation logic 302. When asserted, LATCH DATA indicates that valid data is present on the data portion of ADDRESS, DATA BUSES. Signature generation unit 304 will latch this data and incorporate it into its generated signature. When de-asserted, signature generation unit 304 should ignore the data on the bus. FREEZE is generated by validation logic 302. When asserted, SIM 204 disables the CPU 1 and CPU 2 internal clocks. CPU 1 and CPU 2 cease operations, allowing validation logic 302 to access memory in special circumstances. When de-asserted, CPU 1 and CPU 2 operate normally. SIGNATURE READY is an interrupt generated by validation logic 302. Memory validation logic 214 asserts SIGNATURE READY to indicate the completion of a signature. When de-asserted, one of three conditions has occurred: (1) the interrupt is masked, (2) the signature is not complete, (3) the validation logic 302 is not enabled. SIGNATURE FAILED is an interrupt generated by validation logic 302. Memory validation logic 214 asserts SIGNATURE FAILED to indicate that a completed validation operation has failed. When de-asserted, one of four conditions has occurred: (1) the interrupt is masked, (2) the operation is not complete, (3) the operation is complete but did not fail, or (4) the validation logic 302 is not enabled.

Detailed Description Text (14):
FIG. 4 depicts a programmers model of the CONTROL/STATUS register depicted in FIG. 3. The CONTROL/STATUS register contains eight single-bit control fields and two single-bit status fields: BIT 0 Memory Validation Select (MVS) If MVS is set, then memory validation logic 214 is enabled. If MVS is cleared, then memory validation logic 214 is not enabled. BIT 1 Cycle Steal (CS) If CS is set, then validation logic 302 may assert the control signal FREEZE, forcing a bus access in certain circumstances. If CS is cleared, then validation logic 302 may not assert the FREEZE. BIT 2 Signature Start (SS) If SS is set, then validation logic 302 initiates a validation operation. If SS is cleared, then validation logic 302 remains idled. BIT 3 Automatic Checksum (ACS) If ACS is set, then validation logic 302 automatically compares the output of signature generation unit 304 with the value stored in the CHECKSUM register and stores the result in the CI field. If ACS is cleared, then validation logic 302 does not store the result of the comparison into the CI field. BIT 4 Signature Mask Interrupt (SMI) If SMI is set, then validation logic 302 masks the SIGNATURE READY interrupt. If SMI is cleared, then validation logic 302 outputs the interrupt as generated. BIT 5 Comparison Failed Mask Interrupt (CMI) If CMI is set, then validation logic 302 masks the SIGNATURE FAILED interrupt. If CMI is cleared, then validation logic 302 outputs the interrupt as generated. BIT 6 Loop Control (LC) If LC is set, then validation logic 302 will execute another validation operation after completing a preceding operation. If LC is cleared, then validation logic 302 will only perform a single validation operation. BIT 7 LFSR Reset (LR) If LR is set, then the signature generation unit 304 clears its internal state to a known value. If LR is cleared, then signature generation unit 304 remains in its current state. BIT 8 Signature Interrupt Flag (SI) Validation logic 302 sets this flag to indicate that a signature is available for comparison. CPU 1 clears this bit during its interrupt service routine. BIT 9 Comparison Interrupt Flag (CI) Validation logic 302 sets this flag to indicate that a validation operation is complete. CPU 1 clears this

bit during its interrupt service routine.

Detailed Description Text (17):
Continuing with step 508, if the buses are available or after the completion of step 516, validation logic 302 fetches data stored at the address indicated by the contents of the CURRENT ADDRESS REGISTER, a step 518. Validation logic 302 latches the data into signature generation unit 304 by asserting the control signal LATCH DATA, a step 520. Next, validation logic 302 increments the value stored in the CURRENT ADDRESS register in preparation for the next memory access, a step 522. Validation logic 302 determines if it has completed inspecting the entire range of memory addresses (STOP ADDRESS=ADDRESS BUS value), a step 524. If validation logic 302 has not inspected the entire memory range, then it continues processing at step 506. If validation logic 302 has inspected the entire memory range, then it continues processing at step 526.

Detailed Description Text (18):
In step 526, validation logic 302 indicates the inspection of all memory addresses and the generation of a signature by setting the SI status field. Next, validation logic 302 determines if signature interrupts are allowed (SMI=0), a step 528. If signature interrupts are allowed, then validation logic 302 asserts the control signal SIGNATURE READY, a step 530. If signatures are not allowed or after step 530, validation logic 302 determines if it is to compare the signature output by signature generation unit 304 with the contents of CHECKSUM register (ACS=1), a step 532. If validation logic 302 is to compare the two values, then validation logic performs the comparison and writes the result to the CI status field, a step 534. If the comparison is not enabled or after step 534, validation logic 302 determines if the validation operation is successful (two values match), a step 536. If the operation was successful, then validation logic 302 determines if the loop mode is enabled (LC=1) a step 538. If the loop mode is enabled, then validation logic 302 continues processing at step 506. If the loop mode is not enabled, the operation of validation logic 302 is complete, a step 540.

Detailed Description Text (19):
Returning to step 536, if the validation operation was not successful, then validation logic 302 indicates the failed comparison by setting the CI status field, a step 542. Next, validation logic 302 determines if comparison interrupts are allowed (CMI=0), a step 544. If comparison interrupts are allowed, then validation logic 302 asserts the control signal SIGNATURE FAILED, a step 546. If comparison interrupts are not allowed or after the step 546, then the operation of validation logic 302 is complete (step 540).

US Reference Patent Number (8):
6085334

CLAIMS:

1. A memory validation system comprising: signature generation logic responsive to data retrievable from a bus coupled to a memory during a second access period when the bus is free, where the bus is controllable by a device operating in a standard execution mode during a first access period, the first access period occurring before the second access period, the signature generation logic having an output that produces a first data item that correlates to the data retrieved from the bus; and a comparator responsive to the output of the signature generation logic to compare the first data item to a predetermined second data item.

3. The memory validation system of claim 2, wherein a third data item is retrieved by the signature generation logic during a fourth access period when the bus is free.

7. The memory validation system of claim 4, further comprising a plurality of

registers, the plurality of registers including at least one of a start register, a stop register, a control and status register, a checksum register, a current address register, and a signature register.

10. The memory validation system of claim 9, wherein the signature generation logic receives a third data item from the bus while the freeze signal is asserted.

11. An apparatus comprising: a memory; a bus coupled to the memory; a device coupled to the bus to control the bus during a first time period; a memory validation system including: signature generation logic responsive to data retrievable from the bus during a second time period when the bus is free from control of the device, the signature generation logic having an output to produce a first data item; and a comparator responsive to the output of the signature generation logic to compare the first data item to a predetermined second data item.

16. An electronic control apparatus comprising: an input interface to receive signals from at least one sensor; a digital controller responsive to the input interface, the digital controller including: a memory; a bus coupled to the memory; a device coupled to the bus to control the bus during a first time period; a memory validation system including: signature generation logic responsive to data retrievable from the bus during a second time period when the bus is free from control of the device, the signature generation logic having an output to produce a first data item; and a comparator responsive to the output of the signature generation logic to compare the first data item to a predetermined second data item; and an output interface responsive to the digital controller.

20. A method of processing data to validate memory content, the method comprising: determining whether a bus coupled to a memory is available for access; fetching data from the bus when the bus is available for access; performing signature generation on the data to produce a first data item; and comparing the first data item with a second data item.

23. The method of claim 22, further comprising fetching data from the bus and performing signature generation on the data to produce the first data item after comparing the value of the counting device to the predetermined value.

27. The method of claim 20, wherein signature generation is performed after a plurality of data words are retrieved from the bus.

28. The method of claim 27, wherein signature generation is performed a plurality of different times.

30. The method of claim 20, further comprising retrieving a control bit selected from the group consisting of a memory validate select bit, a cycle steal bit, a signature start bit, an automatic checksum bit, a signature mask interrupt, a comparison failed mask interrupt bit, a loop control bit, a signature interrupt flag bit, a comparison interrupt flag bit, and a signature generation logic reset bit.

Previous Doc Next Doc Go to Doc#

L15: Entry 1 of 1 File: USPT Jul 4, 2000

DOCUMENT-IDENTIFIER: US 6085334 A
TITLE: Method and apparatus for testing an integrated memory device

Detailed Description Text (2):
According to one aspect of the present invention, a method of memory test detects defects in a memory array which are sensitive to environmental conditions. An environmentally sensitive defect is characterized by any change in the repair state of the memory device over the operating range of the memory device. The method generates a signature value to reflect the repair state of the memory device at a first operating condition and generates a second signature value to reflect the repair state of the memory device at a second operating condition. Environmentally sensitive defects are detected when the second signature value is different from the original signature value. Effectively, a memory device is rejected if there is a change in the repair state of the memory array over the operating range of the device.

Detailed Description Text (7):
In another aspect of the invention, a method for testing and repairing an embedded memory in an integrated circuit includes the steps of: (i) setting a first predetermined environmental condition for testing the embedded memory; (ii) activating a built-in self-test circuit and a built-in self-repair circuit to detect and repair faulty memory cells of the embedded memory under the first predetermined environmental condition; (iii) generating a first signature based on addresses of the faulty memory cells determined during testing under the first predetermined environmental condition; (iv) providing the first signature to a first storage location; (v) setting a second predetermined environmental condition for testing the embedded memory; (vi) clearing results of detecting and repairing faulty memory cells under the first predetermined environmental condition; (vii) activating the built-in self-test circuit and the built-in self-repair circuit to detect and repair faulty memory cells of the embedded memory under the second predetermined environmental condition; (viii) generating a second signature based on addresses of the faulty memory cells determined during testing under the second predetermined environmental condition; (ix) providing the second signature to a second storage location; and (x) comparing the first signature to the second signature, and if the first signature is not the same as the second signature, determining that the embedded memory does not meet a predetermined specification.

Detailed Description Text (24):
During production, the repair signature will be captured on the test equipment and may be tracked throughout the device's test. Typically, the test equipment will compare a first signature with a second signature to see if there has been a change. When a signature changes due to environmental conditions, such as voltage or temperature change, the device is rejected.

Detailed Description Text (27):
At block 78, a second environmental condition is set. The second environmental condition may change multiple conditions or may only change one condition. Processing flow continues to block 80 where BIST/BISR testing and repair is

performed once again to detect and repair defects. At the completion of this second BIST/BISR testing and repair cycle, a second repair signature is generated to indicate the failed memory locations discovered during the second BIST/BISR testing and repair cycle. At decision block 83 the first signature is compared to the second signature. Decision block 83 determines if there is a change in the signature. If there is a change in the signature, then the memory is rejected at block 84. If there is no change in the signature, then testing continues at block 86. If the two signatures are the same, testing continues and may include the testing of other modules or a return to block 78 to set another environmental condition.

Previous Doc Next Doc Go to Doc#